J Intell Robot Syst (2009) 56:47–67
DOI 10.1007/s10846-008-9302-x

An Optimal Approach to Collaborative Target Tracking with Performance Guarantees

Jason Derenick · John Spletzer · Ani Hsieh

Received: 15 April 2008 / Accepted: 1 December 2008 / Published online: 6 January 2009

Abstract In this paper, we present a discrete-time optimization framework for target tracking with multi-agent systems. The "target tracking" problem is formulated as a generic semidefinite program (SDP) that when paired with an appropriate objective yields an optimal robot configuration over a given time step. The framework affords impressive *performance guarantees* to include full target coverage (i.e. each target is tracked by at least a single team member) as well as maintenance of network connectivity across the formation. Key to this work is the result from spectral graph theory that states the second-smallest eigenvalue—λ_2—of a weighted graph's Laplacian (i.e. its inter-connectivity matrix) is a measure of connectivity for the associated graph. Our approach allows us to articulate agent-target coverage and inter-agent communication constraints as linear-matrix inequalities (LMIs). Additionally, we present two key extensions to the framework by considering alternate tracking problem formulations. The first allows us to guarantee k-coverage of targets, where each target is tracked by k or more agents. In the second, we consider a relaxed formulation for the case when network connectivity constraints are superfluous. The problem is modeled as a second-order cone program (SOCP) that can be solved significantly more efficiently than its SDP counterpart—making it suitable for large-scale teams (e.g. 100's of nodes in real-time). Methods for enforcing inter-agent proximity constraints for collision avoidance are also presented as well as simulation results for multi-agent systems tracking mobile targets in both $\mathbb{R}^2$ and $\mathbb{R}^3$.

J. Derenick (✉) · J. Spletzer
Computer Science and Engineering, Lehigh University, Bethlehem, PA 18015, USA
e-mail: derenick@lehigh.edu

J. Spletzer
e-mail: josa@lehigh.edu

A. Hsieh
Mechanical Engineering and Mechanics, Drexel University, Philadelphia, PA 19104, USA
e-mail: mhsieh1@drexel.edu

Keywords Optimal target tracking · Convex optimization · Semidefinite programming · Second-order cone programming · Multi-agent systems

1 Introduction

We are interested in developing robot teams for use in surveillance and monitoring applications. The idea of using teams of small, inexpensive robotic agents to accomplish various tasks is one that has gained increasing interest as embedded processors and sensors become smaller, more capable, and less expensive. To this point, much of the work in multi-robot coordination has focused on control and perception. It has generally been assumed that each team member has the ability to communicate with any other member with little to no consideration for the quality of the wireless communication network. Such an assumption, although valid in certain situations, does not generally hold—especially when a team is operating in a highly dynamic environment.

Our previous work in target tracking made similar simplifying assumptions, as no constraints were placed on sensing and communication ranges [1]. This allowed target coverage and network connectivity requirements to be ignored in order to simplify the proposed optimization process. In this paper, however, we consider controlling the configuration of a team of mobile agents for target tracking *under target coverage and inter-agent communication constraints*. Our methodology is based on the graph theoretic result where the second smallest eigenvalue of the inter–connection graph Laplacian matrix is a measure for the connectivity of the graph. Recent system and control literature has shown that the maximization of the second smallest eigenvalue for a state dependent graph Laplacian matrix can be formulated as a semidefinite program [2]. We apply these results to the target tracking task and obtain a coordination strategy that maintains target coverage and network connectivity while minimizing a given objective function over some time step. Specifically, robot-target assignments and inter-agent communication constraints are respectively embedded in *sensor visibility* and *network connectivity* graphs. The target tracking problem is then formulated as a SDP where said constraints are modeled as linear-matrix inequalities (LMIs).

An important advantage of this formulation is that it is agnostic to the quality metric being minimized. If the objective function is convex and constraints can be expressed in linear, quadratic, or semidefinite form, the resulting problem will be convex. Convexity ensures that any solution will be globally optimal and attainable in polynomial time with respect to the number of robots and observation targets.

2 Related Work

In recent years, increased attention has been focused on the effects of communication networks in multi-agent systems. Earlier works generally assumed static communication ranges, [3], and/or relied on coordination strategies that require direct line-of-sight, [4]. In [5] and [6] decentralized controllers were used for concurrently moving toward goal destinations while satisfying communication constraints by maintaining line-of-sight and assuming static communication/sensor ranges respectively. Coordination strategies based on inter-agent signal strength include [7, 8], and [9]. In [10], low-level reactive controllers capable of responding to changes in signal strength or estimated available bandwidth are used to constrain robots' movements in surveillance and reconnaissance tasks. Although much of the recent works have focused on the effects of communication maintenance on navigation, few have addressed the issue of communication maintenance in tasks such as collaborative/collective localization and data fusion where team connectivity is essential to the team's ability to achieve its goals.

Previous works in collaborative target localization include [11, 12], and [13] where strategies such as maintaining visibility constraints and determining optimal sensor placement are considered. More recent works include [14] where artificial potential functions are used to coordinate a team of mobile agents to track multiple moving targets. Jung [15] addresses the same problem by formulating it as two sub-problems: target tracking for a single robot and on-line motion coordination strategy for a team of robots. In [16], the authors consider a motion coordination strategy to enable a team of mobile sensors to detect multiple targets within a given region. [17] analyses the accuracy of cooperative localization and target tracking in a team of mobile robots using an Extended Kalman Filter (EKF) formulation and provide upper bounds for the position uncertainty obtained by the team. In [18], a distributed control strategy is used to maintain a team of mobile agents in a mesh formation to enable tracking of a discrete or diffused target. In [1], the authors employ particle filters to minimize the expected error in tracking mobile target(s) for a team of mobile robots without the use of explicit switching rules, while [19] employs an Extended Kalman Filter (EKF) approach to minimize the position covariance of the targets.

Lastly, maximizing the second smallest eigenvalue (i.e. λ_2) of a state-dependent Laplacian matrix associated with a network graph has been considered in both [2] and [20]. The former is perhaps most related to our work where the problem of finding optimal node positions is formulated as a SDP. Leveraging these results, the latter formulates a fully distributed framework for approximating λ_2.

In contrast to these efforts, we propose a SDP formulation for controlling the configuration of a team of mobile agents for tracking moving targets while maintaining sensing and communication constraints and optimizing an additional tracking objective over a discrete–time step. Furthermore, in situations where one is willing to forgo communication maintenance to ensure complete coverage of all targets, we show how our formulation can be simplified into a SOCP that guarantees each target is tracked by at least a single team member. This is relevant when communication with other team members must be sacrificed to ensure all targets are monitored or when communication ranges far exceed those of on-board sensing modalities.

3 Problem Statement

The objective of this paper is to provide a general framework that facilitates optimal target tracking with performance guarantees. More precisely, we consider minimizing some *convex* objective function, $\Psi : \mathbb{R}^{3n} \to \mathbb{R}$, over a given time step while ensuring

1. Full target coverage (i.e. each target is tracked by at least a single team member)
2. Network connectivity across the robot formation

In this context, Ψ is as a function of our decision variable $X = \left(x_1^a, x_2^a, \ldots, x_n^a\right)^T \in \mathbb{R}^{3n}$ denoting the concatenated positions of the robot team with $x_i^a \in \mathbb{R}^3$ representing the location of agent i with respect to some world frame $\mathcal{W}$. Additionally, we assume a fully actuated motion model for each team member, i.e. where

$$\dot{x} = u, \; u \in \mathcal{U} \subseteq \mathbb{R}^3 \tag{1}$$

Initially, we avoid discussion of Ψ and instead focus on formulating a sufficient constraint set to ensure our desired performance guarantees.

3.1 A State-dependent Graph Representation

Towards this end, we begin by exploiting the fact that a team of robots and the targets they track collectively define a finite, weighted graph where an agent–target edge corresponds to a single point-to-point sensor track. More precisely, letting $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$ and $\mathcal{O} = \{o_1, o_2, \ldots, o_m\}$ respectively denote the full set of agents and the full set of observation targets, we define the graph $G_V(\mathcal{V}_V, \mathcal{E}_V)$ where $\mathcal{V}_V = \mathcal{A} \cup \mathcal{O}$ and $\mathcal{E}_V = \{e \colon e \in \mathcal{A} \times (\mathcal{A} \cup \mathcal{O})\}$. We refer to this graph as the *visibility graph* and associate with its edges a mapping $f_V \colon \mathcal{E}_V \to \mathbb{R}^+$.

Letting $x_j^t \in \mathbb{R}^3$ denote the position of observation target o_j in world coordinate frame $\mathcal{W}$, we define

$$f_V(y) = \begin{cases} f_V^a\left(\| x_i^a - x_j^a \|_2\right), & y = (a_i, a_j) \in \mathcal{A} \times \mathcal{A} \\ f_V^t\left(\| x_i^a - x_j^t \|_2\right), & y = (a_i, o_j) \in \mathcal{A} \times \mathcal{O} \end{cases} \tag{2}$$

where $0 \leq f_V^a(x_i^a, x_j^a),\ f_V^t(x_i^a, x_j^t) \leq 1$. In other words, the weights of the corresponding edges are a direct functional of the relative Euclidean distance separating an agent from some other observable entity. Notice that this also implicitly makes G_V a function of the positional state vector X, and as such, we accordingly denote it $G_V(X)$. Figure 1 illustrates the visibility graph for a team of three agents tracking a single target in $\mathbb{R}^2$.

3.2 Guaranteeing Full Target Coverage

Given the definition of $G_V(X)$, observe that all targets in the system are tracked whenever the graph itself is connected as this implies edges between all targets and at least one member of the agent team. Accordingly, we would like our constraint set to capture this notion of connectivity when determining the optimal state vector X.

With this in mind, we turn our attention to recent results from spectral graph theory regarding the *connectivity* of an arbitrary graph $G(\mathcal{V}, \mathcal{E})$. In particular, we note that the constraint $\lambda_2(L(G)) > 0$ is both a necessary and sufficient condition for

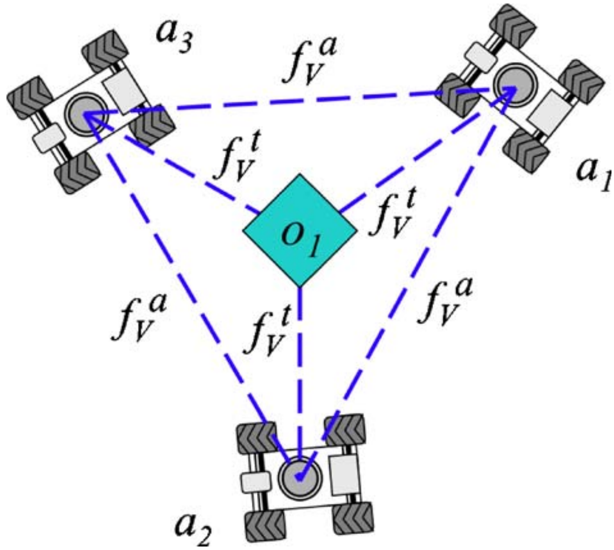

Fig. 1 The weighted visibility graph, $G_V(\mathcal{V}_V, \mathcal{E}_V)$, for a team of three robots observing a single target in $\mathbb{R}^2$. In our formulation, respective edge weights are a function of the team's positional state vector, X

guaranteeing the connectivity of G [21], where $\lambda_2(L(G))$ denotes the second smallest eigenvalue of the weighted graph Laplacian $L(G)$ given by

$$[L(G)]_{ij} = \begin{cases} -w_{ij}, & i \neq j \\ \sum_{i \neq k} w_{ik}, & i = j \end{cases} \tag{3}$$

with w_{ij} being the weight associated with the edge shared between vertices i and j.

In light of these observations, we can now pose the following initial formulation for the target tracking problem

$$\begin{aligned} &\min \Psi(X) \\ &\text{s.t. } \lambda_2(L_V(X)) > 0 \end{aligned} \tag{4}$$

where $L_V(X)$ denotes the state-dependent Laplacian of the visibility graph $G_V(X)$.

Noting the results of [2], we see that

$$\lambda_2(L_V(X)) > 0 \iff P_V^T L_V(X) P_V \succ 0 \tag{5}$$

where $P_V \in \mathbb{R}^{(n+m)\times(n+m-1)}$ comprises an orthonormal basis for an $n+m-1$ dimensional subspace such that $\forall x \in \mathbf{span}(P_V)$, $1^T x = 0$. As such, we can further solidify the problem statement by reposing (4) as follows

$$\begin{aligned} &\min \Psi(X) \\ &\text{s.t. } P_V^T L_V(X) P_V \succ 0 \end{aligned} \tag{6}$$

In this formulation, we adopt the standard Löwner ordering. Additionally, we highlight that *solving this problem minimizes the chosen objective while ensuring that all specified targets are tracked by at least a single team member.*

3.3 Enforcing Network Connectivity Constraints

Although solving (6) yields a positional configuration that will ensure all targets are tracked by at least a single agent, it makes no guarantees regarding the underlying network connectivity over the resulting formation. The ability to ensure a connected network graph while performing such a task is often desirable as it facilitates – among other things – distributed sensor fusion. To address this, we extend our formulation by introducing a network proximity graph $G_N(\mathcal{V}_N, \mathcal{E}_N)$ where $\mathcal{V}_N = \mathcal{A}$ and $\mathcal{E}_N = \{e\colon e \in \mathcal{A} \times \mathcal{A}\}$. Similar to the previous graph formulation, we associate a weight function $f_N\colon \mathcal{E}_N \to \mathbb{R}^+$ that is a direct functional of the Euclidean distance between network peers. For our purposes, f_N will be modeled to reflect the quality of a communication link shared between said nodes.

Given this definition, we can augment (6) accordingly to yield the following problem statement

$$\begin{aligned} &\min \Psi(X) \\ &\text{s.t. } P_V^T L_V(X) P_V \succ 0 \\ &\quad\;\; P_N^T L_N(X) P_N \succ 0 \end{aligned} \tag{7}$$

Solving this problem, will yield an optimal positional configuration for the team that maintains both network connectivity as well as complete target coverage. However, it should be noted that (7) is not necessarily a convex optimization problem. As we shall see, this does not prevent us from formulating the target tracking problem as a discrete-time process whereby during each iteration a convex form of (7) is solved.

4 Defining Interactive Control Functions

Given this formulation, we now consider appropriate definitions of f_V^a, f_V^t, and f_N. The choice of these functions is critical as they inherently govern the behavior of the team. As these functions dictate the relationship between one node and another as well as any observation targets, we see that at the highest level that they can be considered *interactive control* functions. With this in mind, we now consider appropriate choices for a simple target tracking scenario.

Momentarily neglecting discussion of f_V^a, we focus instead on characterizing the desirable properties of weight function f_V^t, which governs the interactions of systems agents and respective targets. In an ideal tracking scenario the team will have an optimal number of tracks while each agent maintains a safe standoff distance between itself and its targets. In other words, letting r_{ij} denote the desired distance between agent a_i and target o_j, we would like f_V^t to promote the team to behave such that

$$\left|r_{ij} - \epsilon_{ij}^l\right| \leq d_{ij}^t \leq \left|r_{ij} + \epsilon_{ij}^u\right|, \forall i, j \tag{8}$$

where $d_{ij}^t \triangleq \| x_i^a - x_j^t \|_2$ and $\epsilon_{ij}^l, \epsilon_{ij}^u \in \mathbb{R}^+$ dictate the acceptable lower/upper bound tolerances for any tracks between agent a_i and observation target o_j. When the right-hand inequality holds, we define the track as being *active*.

Seeking inspiration from the motion-planning community, we opt to consider a formulation of f_V^t modeled after common potential functions. For our purposes, we consider the case $\epsilon_{ij}^l = \epsilon_{ij}^u$ and model f_V^t using a symmetric Gaussian potential – noting that the subsequent analysis can easily be adapted for an alternate (symmetric or non-symmetric) potential function formulation. That stated, we consider the following definition

$$f_V^t\left(x_i^a, x_j^t\right) = \begin{cases} e^{-\frac{\gamma^2}{2}\left(d_{ij}^t - r_{ij}\right)^2}, & \left|d_{ij}^t - r_{ij}\right| \leq \epsilon \\ 0, & otherwise \end{cases} \tag{9}$$

where γ is a "strictness" parameter defined as function of ϵ—e.g. we've used $\gamma = \sqrt{\frac{7}{\epsilon}}$. Figure 2 (Left) illustrates this function for an instance with $r_{ij} = 10$ and $\epsilon = 4$.

Our choice of f_V^t has two primary motivations. The first is the differentiability properties of the Gaussian which facilitates problem formulation. The second relates to our operational objectives for tracking. In many sensor systems, position tracking errors are proportional to the standoff distance. For example, in stereo vision the propagation of errors in disparity measurements is proportional to the square of the scene depth. When fusing bearing measurements from a pair of cameras on two different agents, the error is directly proportional to the standoff distances [22]. Thus, we desire each agent to be as close to the target as possible to enhance tracking accuracy, while still maintaining a sufficient standoff distance to avoid detection and/or collision. This desired behavior is implicitly captured in our choice of f_V^t.

Regarding f_V^a, which governs inter-agent tracking behaviors, we see its definition is not as obvious since a variety of formulations may lead to favorable results depending upon the chosen application and mission objectives. For instance, we can choose $f_V^a = 1$ which has the effect of removing any inter-agent observability requirements as it essentially says that no matter what the positional state of the team, the inter-agent links are connected or are observable everywhere. Whether this is feasibly possible given the agents' respective sensor suites is inconsequential for the task at hand, as in choosing the weights this way we are only concerned with ensuring complete target coverage. Another reasonable choice for f_V^a is a potential function that behaves similarly to that used to define f_V^t. This definition would be

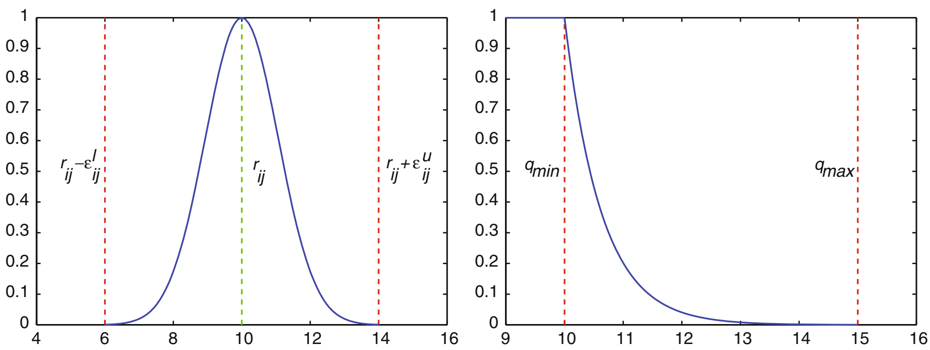

Fig. 2 (*Left*) An instance of f_V^a where $r_{ij} = 10$ and $\epsilon_{ij}^l = \epsilon_{ij}^u = 4$, $\forall i, j$. For our purposes, a simple symmetric Gaussian was utilized to govern agent-target interactions; however, a different potential function could have just as easily been used. (*Right*) An instance of a simple exponential-decay model for RF links with $q_{min} = 10$ and $q_{max} = 15$

useful in a scenario where team members rely upon local observations of their peers for such things as localization. In this paper, we adopt the former definition.

In a similar manner, we can now address the issue of weighting the network links in $G_N(X)$. The weights of these links are very easily characterized, and such a formulation has been addressed in recent literature [2, 20]. For our purposes as well as for the sake of further discussion, we consider the exponential decay model posed by [20]. Doing so yields the following formulation for f_N

$$f_N\left(x_i^a, x_j^a\right) = \begin{cases} 1, & d_{ij}^a \le q_{min} \\ e^{\frac{-5(d_{ij}^a - q_{min})}{q_{max} - q_{min}}}, & q_{min} < d_{ij}^a \le q_{max} \\ 0, & d_{ij}^a > q_{max} \end{cases} \tag{10}$$

Figure 2 (Right) shows a single instance of f_N for $q_{min} = 10$ and $q_{max} = 15$.

5 Defining a Discrete Semi-definite Approach

In this section, we consider formulating our problem as a discrete–time process whereby the agent team collectively observes the relative positions of the observation targets and then accordingly adjusts their respective trajectories so as to minimize the given objective. As the targets are assumed dynamic and control is inherently a discrete–time process, we see that at best the team can only optimize Ψ over the period Δt representing the rate at which they are able to effectively sample the environment and issue control signals. Although this approach does not guarantee optimally-convergent behavior for the team, it *does ensure* that the solution obtained will yield a trajectory that is optimal with respect to Ψ over that time step.

5.1 Problem Formulation

With this in mind, we leverage the results of [2] who considered a discrete-time process for maximizing network connectivity in multi-agent teams. Following suit, we perform a simple differentiation with respect to time and then apply Euler's first-order discretization method. Doing so reveals the following discrete-time representation of f_V^t

$$\begin{aligned} f_V^t\big(x_i^a, x_j^t\big)(k+1) - f_V^t\big(x_i^a, x_j^t\big)(k) &= \tau_{V_k}^t \left\{x_i^a(k) - x_j^t\right\}^T \Delta x_i^a \\ \tau_{V_k}^t &= -\frac{\gamma^2\big(\left\|x_i^a(k) - x_j^t\right\|_2 - r_{ij}\big)}{\left\|x_i^a(k) - x_j^t\right\|_2} f_V^t\big(x_i^a(k), x_j^t\big) \end{aligned} \tag{11}$$

where $\forall l, a_l \in \mathcal{A}$, we have $\Delta x_l^a = x_l^a(k+1) - x_l^a(k)$.

Similarly for f_N, we obtain

$$f_N\big(x_i^a, x_j^a\big)(k+1) - f_N\big(x_i^a, x_j^a\big)(k) = \tau_{N_k} \left\{x_i^a(k) - x_j^a(k)\right\}^T \left\{\Delta x_i^a - \Delta x_j^a\right\}$$

$$\tau_{N_k} = -\frac{5\, f_N\big(x_i^a(k), x_j^a(k)\big)}{(q_{max} - q_{min})\left\|x_i^a(k) - x_j^a(k)\right\|_2}$$

Given (11) and (5.1) and recalling f_V^a is chosen constant (i.e. $f_V^a(x_i^a, x_j^a)(k+1) - f_V^a(x_i^a, x_j^a)(k) = 0$), we can now define the discretized state-dependent Laplacians with respect to both the visibility graph $G_V(X)$ and the network proximity graph $G_N(X)$. For the former, we obtain

$$\left[L_V(k+1)\right]_{uv} = \begin{cases} -f_V(y_u, y_v)(k), & u \neq v \\ \sum\limits_{u \neq s} f_V(y_u, y_s)(k), & u = v \end{cases} \tag{12}$$

where y_u and y_v are defined such that

$$y_l = \begin{cases} x_l^a, & l \leq n = |\mathcal{A}| \\ x_{(l-n)}^t, & n < l \end{cases}$$

Similarly for $G_N(X)$, we are able to define

$$\left[L_N(k+1)\right]_{uv} = \begin{cases} -f_N\left(x_u^a, x_v^a\right)(k), & u \neq v \\ \sum\limits_{u \neq s} f_N\left(x_u^a, x_s^a\right)(k), & u = v \end{cases} \tag{13}$$

Putting this all together, we arrive at a discrete-time formulation for optimal target tracking. At time step k, we aim to solve the following problem

$$\begin{aligned} &\min \Psi(X(k+1)) \\ &\text{s.t.} \parallel x_i^a(k+1) - x_i^a(k) \parallel_2 \leq v_i \Delta t, \; i = 1, \ldots, n \\ &\qquad P_V^T L_V(k+1) P_V \succ 0 \\ &\qquad P_N^T L_N(k+1) P_N \succ 0 \end{aligned} \tag{14}$$

where v_i denotes the translational velocity of agent a_i.

It should be noted that we have augmented the problem with n second-order conic inequalities constraining the distance each agent can travel in a single step. These constraints are essential as they serve to reduce the effects of the linearization process. Additionally, they can be used to model velocity constraints on the individual robots. Noting that $\Psi(X(k+1))$ is assumed convex and our feasible set is characterized by LMIs and second-order conic inequalities, we see that (14) is a semidefinite program and can be efficiently solved using interior-point methods for convex analysis [21].

5.2 Choosing an Appropriate Objective

Until this point, we have avoided any detailed discussion regarding the statement of our convex objective, Ψ. In fact, in the context of target tracking there are many useful candidate functions that fit well within this framework. One possibility is to choose Ψ as the trace of the covariance representing the uncertainty in measured target positions [23]. As such, (14) would yield a position vector that minimizes the uncertainty in the estimated target positions while ensuring full target coverage and network connectivity.

In this paper, we instead choose $\Psi(X) = -\lambda_2(L_V(X))$. Given this function, we can then maximize the second smallest eigenvalue of the state-dependent graph Laplacian associated with our visibility graph over a given time step. Choosing Ψ this way aims to maximize the visibility of observation targets. Such an objective may be quite useful in surveillance applications where each member of the mobile team is outfitted with a low-grade sensor suite. In such a case, maximizing visibility while observing network connectivity will provide maximal redundancy in the observation network.

Other appropriate objective functions (e.g. weighted least-squares) could be imagined. However, what is important to note is the generality of our approach. So long as the objective function is convex, and can be expressed in terms of linear, quadratic, or semidefinite constraints, the resulting problem will be a semidefinite program.

5.3 Simulation Results

In an effort to validate our discrete-time framework, we implemented our paradigm in Matlab using SeDuMi 1.1R3 [24] via YALMIP [25]. Figure 3 illustrates the results from one such trial in which the objective was to maximize connectivity in the visibility graph, i.e. $\Psi(X) = -\lambda_2(L_V(X))$. In this scenario, eight networked agents were responsible for tracking five mobile targets while maintaining a desired standoff distance $r_{ij} = 0.10$. The minimal desired agent-target proximity bound for active tracks was set at 0.06 with a maximum of 0.14. In this case, each agent was modeled using as its a primary sensor an omnidirectional camera system, and the network was modeled to experience exponential decay between 0.08 and 0.18 units. The maximal translational velocity of respective team members was 1.4 times that of the targets. Given the contrived trajectories of the targets, the team ultimately converges to an optimal configuration—yielding $\Psi(X) \approx 0.6972$.

Figure 4 reveals the progression of both $\lambda_2(L_V(X))$ and $\lambda_2(L_N(X))$ as the team employs our discrete-time framework. In this plot, we highlight the monotonically increasing behavior of the objective while noting that $\lambda_2(L_N(X))$ remains positive for the entire run. In other words, the team effectively optimizes its objective over each time step while maintaining network connectivity across the formation.

6 Guaranteeing *k*-Coverage of Observation Targets

In Section 3, we formulated the tracking problem to ensure that each target is tracked by at least a single agent; however, there are many scenarios in which it would be desirable to enforce a lower-bound $k_j \in \mathbb{Z}^+$ on the number of agents tracking each target o_j. For instance, teams of low-cost robots may employ sensors incapable of directly estimating the target's positions without additional constraints—e.g. a minimum of two bearing sensors such as cameras are needed to estimate a target's pose.

6.1 Considering an Alternate Formulation

Towards this end, we propose an alternate formulation of (14) by introducing a separate state-dependent visibility graph, $G_{V_j}(X)$, for each o_j. Unlike the visibility graph presented in Section 3.1, the vertices of $G_{V_j}(X)$ correspond to the full set of $q = \binom{n}{n-k_j+1}$ agent combinations (denoted $\mathcal{C}_j = \{\mathcal{A}_1, \ldots, \mathcal{A}_q\}$) with an additional vertex corresponding to o_j. At first glance, this choice seems unintuitive; however—as we shall see—it is in defining the vertices this way that facilitates this key result.

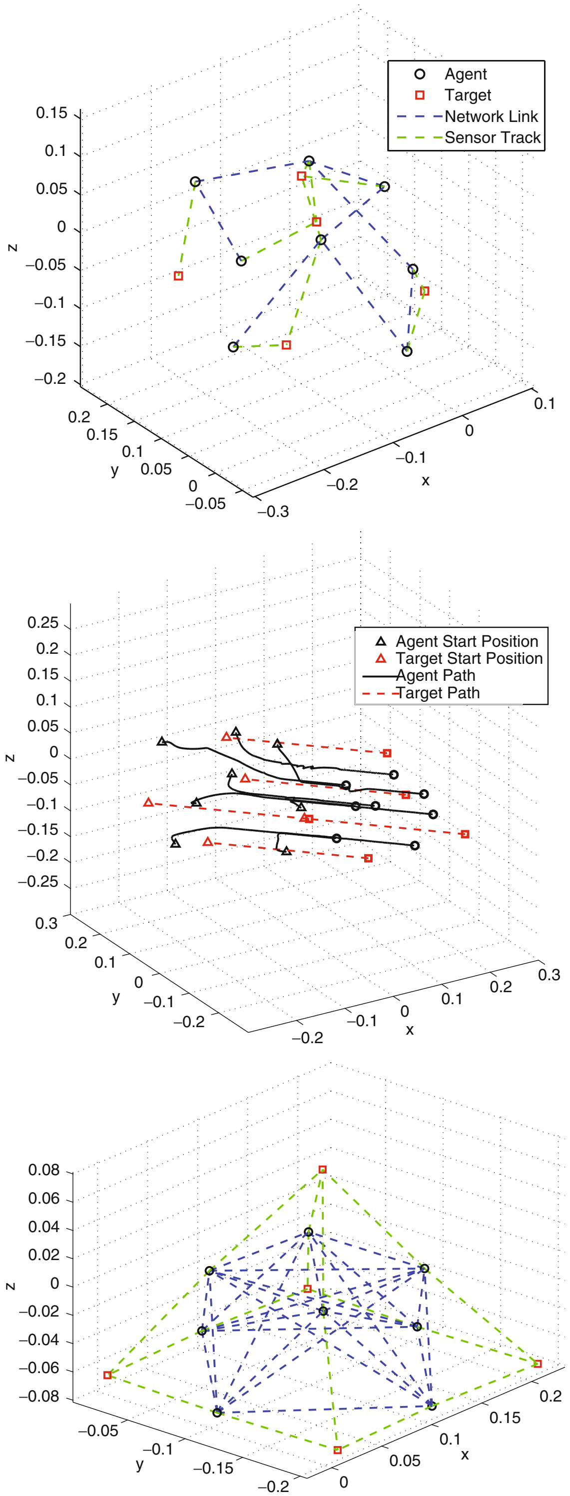

Fig. 3 (*Top*) The initial visibility and network proximity graphs for a team of eight agents in $\mathbb{R}^3$ charged with tracking five mobile targets. In this case, each agent was modeled as using an on-board omnidirectional camera system with the team's objective being to maximize the total number of active tracks—i.e. $\Psi(X) = -\lambda_2(L_V(X))$. (*Middle*) The trajectories of the respective team members as they obtain an optimal configuration. (*Bottom*) The resulting visibility and network graphs for the team after convergence ($\Psi(X) \approx 0.6972$)

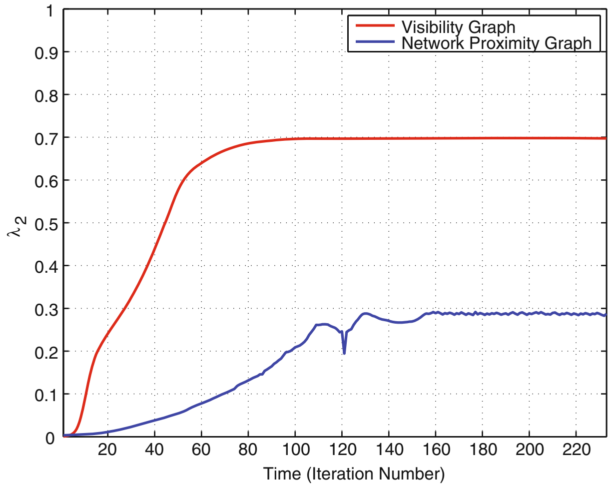

Fig. 4 The progression of $\lambda_2(L_V(X))$ and $\lambda_2(L_N(X))$ corresponding to the run illustrated in Fig. 3. In this case, the objective was to maximize the number of active tracks in the agent configuration over each time step—i.e. $\Psi(X) = -\lambda_2(L_V(X))$. Given the contrived target trajectories, the connectivity of the visibility graph increases monotonically as the team converges to a configuration that maximizes the objective—yielding $\Psi(X) \approx 0.6972$

Momentarily deferring discussion of this point, we now define the corresponding edge set $\mathcal{E}_{V_j} = \mathcal{C}_j \times \{o_j\}$ and associate with each edge a weight function $f_{V_j}^{\mathcal{A}_i} : (\mathcal{A}_i, o_j) \rightarrow \mathbb{R}^+$. For our purposes, $f_{V_j}^{\mathcal{A}_i}$ is a direct functional of the Euclidean distance separating o_j from each agent in the i^{th} combination $\mathcal{A}_i$. If some agent in $\mathcal{A}_i$ is able to observe the target o_j (i.e. it is proximal to the target location), we require $f_{V_j}^{\mathcal{A}_i} > 0$. If all agents in $\mathcal{A}_i$ are unable to observe the target (i.e. none are proximal), we require $f_{V_j}^{\mathcal{A}_i} = 0$ to indicate the corresponding edge is disconnected.

Taking the given vertex and edge sets as well as the chosen weighting functions, we have provided a general definition of $G_{V_j}(X)$. To solidify our formulation, we

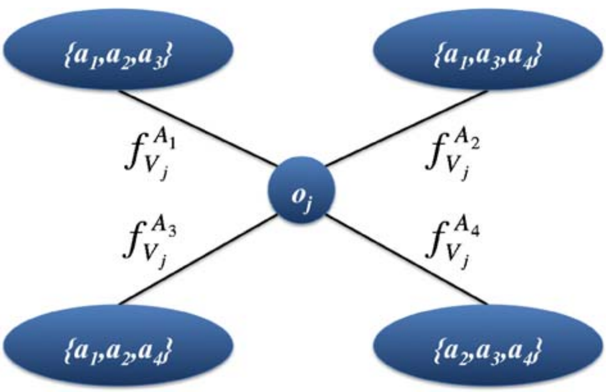

Fig. 5 The visibility graph, $G_{V_j}(X)$, for a team of four agents $\mathcal{A} = \{a_1, \ldots, a_4\}$ tasked with maintaining at least $k_j = 2$ coverage of target o_j. The vertex set is given by of the full set of agent combinations (i.e. $\mathcal{C}_j$) with an additional vertex for o_j. Each edge is associated with a weight function $f_{V_j}^{\mathcal{A}_i}$. Observe that when $f_{V_j}^{\mathcal{A}_i} > 0, i = 1, \ldots, 4$ the graph is connected (i.e. $\lambda_2(G_{V_j}) > 0$) implying that at least two agents are observing the target

present Fig. 5. This figure illustrates $G_{V_j}(X)$ for a team of four agents tasked with ensuring $k_j = 2$ coverage of target o_j. Notice that when $f_{V_j}^{\mathcal{A}_i} > 0, i = 1, \ldots, 4$ the graph is connected implying that at least two agents are observing the target.

Having presented Fig. 5, we now generalize this result with Theorem 1.

Theorem 1 *Let $G_{V_j}(X)$ denote the state-dependent visibility graph associated with target o_j as previously formulated. Assume $f_{V_j}^{\mathcal{A}_i} > 0, i = 1, \ldots, \binom{n}{n-k_j+1}$. (i.e. assume $G_{V_j}(X)$ is connected) then at least $k_j \in \mathbb{Z}^+$ agents are tracking observation target o_j.*

Noting that the $G_{V_j}(X)$ is connected, it is implied that at least a single agent from each of the $\binom{n}{n-k_j+1}$ combinations in C_j is able to observe o_j. Furthermore, observing that agent $a_i \in \mathcal{A}$ appears in exactly $\binom{n-1}{n-k_j}$ combinations in C_j, we see that no more than this number of edges can be affected by a_i's ability to observe target o_j. As a result, proving our minimal coverage bound is equivalent to showing that the ratio of total agent combinations to the number of combinations containing a_i is greater than or equal to k_j. More precisely, proving the following implicitly establishes Theorem 1:

$$\frac{\binom{n}{n-k_j+1}}{\binom{n-1}{n-k_j}} \geq k_j \tag{15}$$

In light of this observation, we now offer the following proof of (15):

Proof By contradiction. Assuming contrary to (15), we see

$$\begin{aligned}
\frac{\binom{n}{n-k_j+1}}{\binom{n-1}{n-k_j}} < k_j &\Rightarrow \frac{n!(n-k_j)!k_j!}{(n-k_j+1)!(k_j-1)!(n-1)!} < k_j \\
&\Rightarrow \frac{nk_j}{n-k_j+1} < k_j \\
&\Rightarrow n < n - k_j + 1 \\
&\Rightarrow k_j < 1 \\
&\Rightarrow k_j \notin \mathbb{Z}^+
\end{aligned}$$

This yields a contradiction from which we conclude that our assumption is false. □

Having established our minimal coverage bound, we now turn our attention to defining an appropriate set of weighting functions for $G_{V_j}(\mathrm{X})$ that satisfy our previously stated criteria. Although there are a variety of functions that we can consider, we opt to extend our previous results and define edge weights as follows

$$f_{V_j}^{\mathcal{A}_i} = \frac{\sum_{s=1}^{n-k_j+1} f_V^t(x_s^a, x_j^t)}{n-k_j+1}$$

where x_s^a and x_j^t respectively denote the positions of $a_s \in \mathcal{A}_i$ and target o_j.

Observe that when target o_j is being actively tracked by all agents in $\mathcal{A}_i$ with each agent observing its desired standoff distance, we have $f_{V_j}^{\mathcal{A}_i} = 1$. Similarly, when no agent in $\mathcal{A}_i$ is engaging the target, we have $f_{V_j}^{\mathcal{A}_i} = 0$.

Accordingly, we now restate (7) as follows

$$\begin{aligned} &\min \Psi(X) \\ &\text{s.t. } P_{V_i}^T L_{V_i}(X) P_{V_i} \succ 0, \; i = 1, \ldots, m \\ &\qquad P_N^T L_N(X) P_N \succ 0 \end{aligned} \tag{16}$$

where $L_{V_i}(X)$ is the state–dependent graph Laplacian associated with $G_{V_j}(X)$, $P_{V_i} \in \mathbb{R}^{(n-k_j+2)\times(n-k_j+1)}$ comprises an orthonormal basis for an $n - k_j + 1$ dimensional subspace such that $\forall x \in \textbf{span}(P_{V_i})$, $1^T x = 0$, and P_N is as previously defined in (7).

Taking this formulation and applying a first–order Euler discretization with respect to time yields the following:

$$\begin{aligned} &\min \Psi(X(k+1)) \\ &\text{s.t.} \left\| x_i^a(k+1) - x_i^a(k) \right\|_2 \le v_i \Delta t, \; i = 1, \ldots, n \\ &\qquad P_{V_i}^T L_{V_j}(k+1) P_{V_i} \succ 0, \; j = 1, \ldots, m \\ &\qquad P_N^T L_N(k+1) P_N \succ 0 \end{aligned} \tag{17}$$

Once again, we've transformed the problem into a SDP yielding a convex formulation that *guarantees each target is observed by a desired minimal number of agents and that the communication network remains connected across the team formation over the given time step*. However, it should be noted that the utility of this approach is hindered by the fact that it introduces m graphs—each having a combinatoric number of edges. As a result, this method is best suited for problems featuring modest k-coverage requirements or a small to moderate number of system agents and targets. However, for multi-agent systems employing cameras or range sensors where 2-coverage is required for tracking, the approach can still be run in real-time.

6.2 Simulation Results

In an effort to validate this approach, we implemented our paradigm in Matlab for a team of four agents charged with tracking two diverging targets while ensuring each is covered by at least $k_{\{1,2\}} = 2$ team members at a requested standoff. Each agent was modeled as having an omnidirectional camera allowing it to track within 0.11 units with the requested standoff distance being 0.07 units. Figure 6 (Left) shows the resulting trajectories for the team. In this example, the objective was to maximize the minimum connectivity of each of the visibility graphs—i.e. $\psi(X) = \min_{j\in\{1,2\}} \{-\lambda_2(L_{V_j}(X))\}$.

Figure 6 (Right,Top) shows the progression of $\lambda_2(L_{V_1}(X))$ with respect to time. As $\lambda_2(L_{V_1}(X)) > 0$, we see that o_1 is guaranteed to be tracked by at least $k_1 = 2$ agents. Figure 6 (Right, Bottom) shows the approximate agent-target distances (i.e. d_{11}^t, d_{21}^t) with respect to time. In this example, both agents readily maintain their active tracks at the requested standoff. The agent velocities were set at 1.125 times that of the observation target velocities, which were assumed equal.

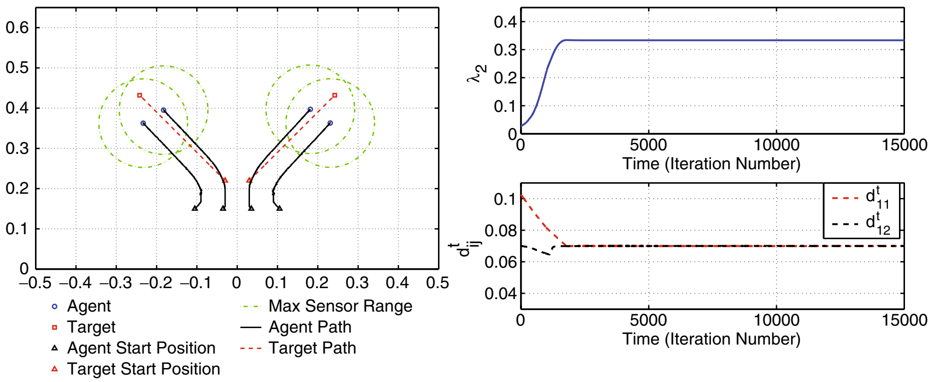

Fig. 6 (*Left*) A team of four agents tracks two diverging targets while ensuring each target is observed by at least $k_{\{1,2\}} = 2$ team members. In this scenario, the objective was to maximize the minimal connectivity of each of the visibility graphs. (*Right*, *Top*) The algebraic connectivity of the $G_{V_1}(X)$ increases monotonically until reaching the value of $\lambda_2(L_{V_1}(X)) \approx 0.3333$. (*Right*, *Bottom*) The relative agent-target distances (i.e. d^t_{11}, d^t_{21}) yielded from maximizing $\lambda_2(L_{V_1}(X))$—both agents maintain the desired standoff of 0.07 units

7 Integrating Inter-agent Collision Avoidance

Until this point, we've made no mention on how to effectively guard against inter-agent collisions. As it turns out, embedding said constraints into our framework is fairly straight–forward. One approach is to follow the lead of [2], who utilized a Euclidean Distance Matrix (EDM) to ultimately enforce the non-convex constraints $\| x^a_i - x^a_j \|_2 \geq \alpha$, $i = 1, \ldots, (n-1)$, $j = (i+1), \ldots, n$. This matrix captures the squares of the distances separating each system agent. Through linearization it can be used as part of an LMI to enforce said bounds. This is the technique we employed in enforcing our inter-agent proximity constraints when generating the results that yielded Fig. 6.

An alternate approach is to incorporate an *inter-agent proximity graph* $G_P(\mathcal{V}_P, \mathcal{E}_P)$ where $\mathcal{V}_P = \mathcal{A}$ and $\mathcal{E}_P = \{e \colon e \in \{\mathcal{A} \times \mathcal{A}\} / \{(a_i, a_i) \colon a_i \in \mathcal{A}\}\}$. Similar to the previous graph formulations, we can associate a weight function $f_P \colon \mathcal{E}_P \to \mathbb{R}^+$ that is also a direct functional of the Euclidean distance separating the system agents. In this case, f_P should be used to approximate an indicator function—showing an edge between two vertices to be connected if and only if the associated agents are "safely" apart.

8 A SOCP Relaxation for Large-scale Teams

In some cases, it may be beneficial to sacrifice connectivity of the underlying network proximity graph in order to ensure each target is tracked. In others, the range of communication links may far exceed the sensing range of the mobile robot team. In such scenarios, the constraint $P_N^T L_N(X) P_N \succ 0$ is superfluous and can be safely eliminated from the problem statement. In doing so, we obtain the original problem formulation presented in (6), which we claim can be effectively relaxed as a SOCP.

8.1 Considering A Relaxed Formulation

The key to obtaining this result is observing that the constraint $P^T L(X) P \succ 0$ reduces to a single non–linear inequality when the graph in question features only a single pair of nodes. As such, we consider a relaxed formulation of the tracking problem whereby we associate with each target o_j a single bi-nodal graph $G_{V_j}(\mathcal{V}_j, \mathcal{E}_j)$ with one vertex serving to represent the agent team (i.e. $\mathcal{A}$) and the other representing the target itself. *By enforcing the connectivity of each of these graphs in our problem formulation, we ensure at least a single active track to each observation target*.

Implicit in this statement is that an appropriate weight function can be formulated for $G_{V_j}(X)$ that fully captures the level of connectivity between the agent team and target o_j. Although a variety of functions can be considered, we extend upon our previous analysis and propose the following

$$f_{V_j}\left(x_1^a, x_2^a, \ldots, x_n^a, x_j^t\right) = \frac{\sum_{i=1}^{n} f_V^t\left(x_i^a, x_j^t\right)}{n} \tag{18}$$

Notice that by this definition, when target o_j is being actively tracked by all network agents with each agent observing its desired standoff distance, we have $f_{V_j} = 1$. Similarly, when no agent is actively engaging the target, we have $f_{V_j} = 0$.

In light of these results, we restate (6) in the following relaxed form

$$\begin{aligned} &\min \Psi(X) \\ &\text{s.t. } P^T L_{V_j}(X) P > 0, \; j = 1, \ldots, m \end{aligned} \tag{19}$$

where $P = [1, -1]^T$.

Once again applying Euler's first-order discretization method, we obtain the following discrete–time formulation

$$\begin{aligned} &\min \Psi(X(k+1)) \\ &\text{s.t.} \left\| x_i^a(k+1) - x_i^a(k) \right\|_2 \leq v_i \Delta t, \; i = 1, \ldots, n \\ &\qquad P^T L_{V_j}(k+1) P > 0, \; j = 1, \ldots, m \end{aligned} \tag{20}$$

This is a standard SOCP constrained by n second-order conic inequalities along with m linear inequalities. It is readily solvable using standard SOCP techniques that are *significantly* more efficient than SDP approaches [21].

8.2 Simulation Results

To illustrate the effectiveness of this novel extension, we implemented the discrete-time SOCP in simulation. Figure 7 (Left) shows the results of one trial where a team of three robots operating in $\mathbb{R}^2$ breaks an initial path formation in order to successfully track three evading targets. Although contrived, this example serves to highlight the governing behavior of our paradigm. By ensuring the connectivity of the $m = 3$ bi-nodal visibility graphs, we see the team is able to ensure full target coverage.

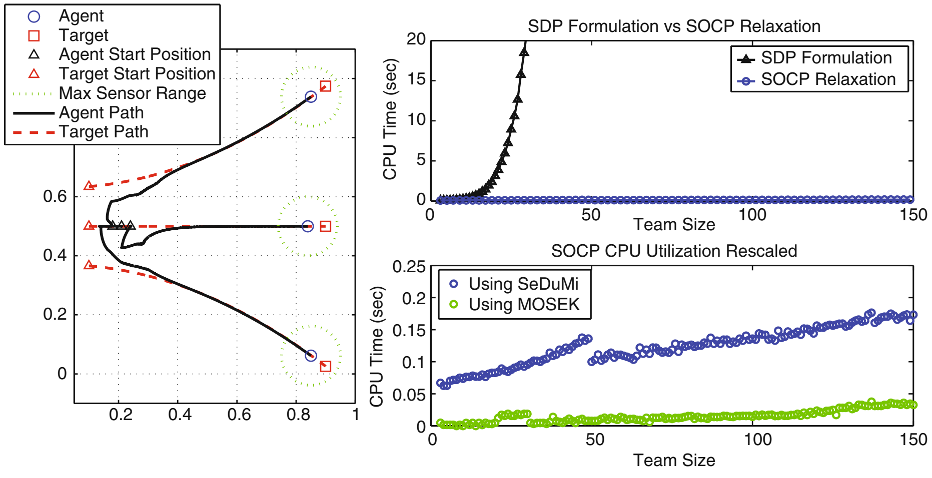

Fig. 7 (*Left*) A team of three agents initially deployed in a path break formation to ensure active tracks of three mobile targets. Connectivity of the state-dependent visibility graph, $G_V(X)$, is ensured by using the proposed SOCP relaxation. (*Right*, *Top*) CPU time obtained from solving both the SDP formulation and SOCP relaxation via SeDuMi for teams having up to 150 agents tracking 3 targets in $\mathbb{R}^2$. (*Right*, *Bottom*) CPU time trends for solving the relaxed problem using both a non-industrial (SeDuMi) and industrial solver (MOSEK)

In this case, the objective was to maximize the minimal respective connectivity of these graphs with $r_{ij} = 0.06$, and $\epsilon^l_{ij} = \epsilon^u_{ij} = 0.04$, $\forall i, j$.

Performance in Practice In an effort to gauge the comparative difference in complexity between the two approaches, we solved instances of (14) and (20) for team sizes up to 150 nodes operating in $\mathbb{R}^2$. In our SDP implementation, we considered the objective $\Psi(X) = -\lambda_2(L_V(X))$ and maintained $f^a_V = 1$. In the SOCP implementation, we considered maximizing the minimal connectivity among the $m = 3$ bi-nodal visibility graphs. In both cases, SeDuMi was used as the underlying solver. All computations were done on a standard desktop computer having a 2.4 GHz Core 2 Duo Pentium Processor with 2 GB RAM. Figure 7 (Right, Top) shows the results of these trials where each data point corresponds to the mean utilization time obtained from solving ten random problem instances.

Not surprisingly, the computational overhead associated with solving the SDP formulation scales cubicly in time. In contrast, the computational load incurred by solving our SOCP relaxation exhibits highly linear growth ($r^2 = 0.9205$). Using SeDuMi, which is a non-industrial grade solver, we see that solving a single iteration of (20) for a team of 150 agents requires 174 ms. In practice, however, it is far more likely that an industrial grade solver will actually be used. As such, we also solved our SOCP relaxation considering the same random problem instances using the MOSEK industrial solver package [26]. The results of these trials are shown in Fig. 7 (Right, Bottom). Again, the computational overhead exhibits an approximately linear trend (in this case, $r^2 = 0.7435$). However, what is perhaps more impressive is that *solving a single iteration of* (20) *for a team of 150 nodes requires only 33 ms!*

9 Towards Characterizing Optimality

As mentioned earlier, the proposed framework guarantees optimal solutions for a given time step. Such a paradigm is ideal for operating in highly dynamic environments where a team can at best optimize over the time it takes for it to sense its surroundings and issue corresponding control signals. However, we are also interested in characterizing the performance of our discrete-time approach in scenarios featuring static target placements. Specifically, we aim to address whether the team will exhibit convergence to global optimality in such cases. Key to this analysis is the realization that each iteration of our process provides a desired descent direction and an implicit step-length towards some optimal team configuration with respect to the current target arrangement. Accordingly, we consider a contrived pair of static target arrangements that allow global solutions to be computed offline given geometric constraints. Although results presented in this section are preliminary, they do highlight the ability of our framework to "seek" a level of optimality extending beyond the current time step.

In the first scenario, a team of ten agents was charged with observing a single target in $\mathbb{R}^2$ while maximizing connectivity of the visibility graph (i.e. $\Psi(X) = -\lambda_2(L_V(X))$ subject to network connectivity constraints. Initial agent positions were randomly chosen with $x_i^a \sim N(0, \sigma_x = \sigma_y = 0.15)$ centered around the target's location. Such a scenario mimics a surveillance team being deployed to a region based upon the believed position of the observation target. As the desired standoff distance for each member was 0.30 units (with $\epsilon_{ij}^l = \epsilon_{ij}^u = 0.15, \forall i, j$), any globally optimal configuration was required to lie on the circle having that radius centered at the target. A total of ten trials were run with each ultimately converging to a globally optimal configuration—yielding $\Psi(X) \approx 1$. Network links were modeled to experience exponential decay between 0.05 and 0.45 units. In our implementation, the process terminated when $\Delta\Psi(X) = \Psi(X(k+1)) - \Psi(X(k)) < 1 \times 10^{-9}$. Figure 8 (Left) illustrates the convergent behavior exhibited during one of the random trials.

In the second scenario, a team of four agents was charged with tracking five targets arranged in a static "cross" formation. Once again, the objective was to maximize target visibility. In this configuration, each target was separated by 0.35 units along the x and y axes of the coordinate frame. Given the desired standoff distance of 0.175 units (with $\epsilon_{ij}^l = \epsilon_{ij}^u = 0.125, \forall i, j$), the globally optimal configuration corresponded to the midpoints along the adjoining line segments of the cross. Initial agent positions were chosen according to the random model employed for the first scenario with the mean corresponding to the center of the target arrangement. Both the same network model and termination criteria were utilized for these trials as well. Of the ten instances, eight resulted in convergence to global optimality—yielding $\Psi(X) \approx 0.382$. In the remaining cases, locally optimal configurations were attained. This local convergence can be attributed to initial arrangements that placed a majority of the team in close proximity to a single periphery target—leaving one target well-beyond sensing range. Such behavior is intuitive as the control functions governing agent–target behaviors are likely to have reduced influence on an agent's position when the target lies outside the exponential well that models sensor visibility. Figure 8 (Right) illustrates the globally convergent behavior attained from one of these trials. Expanding these preliminary results to include more diverse scenarios as well as to

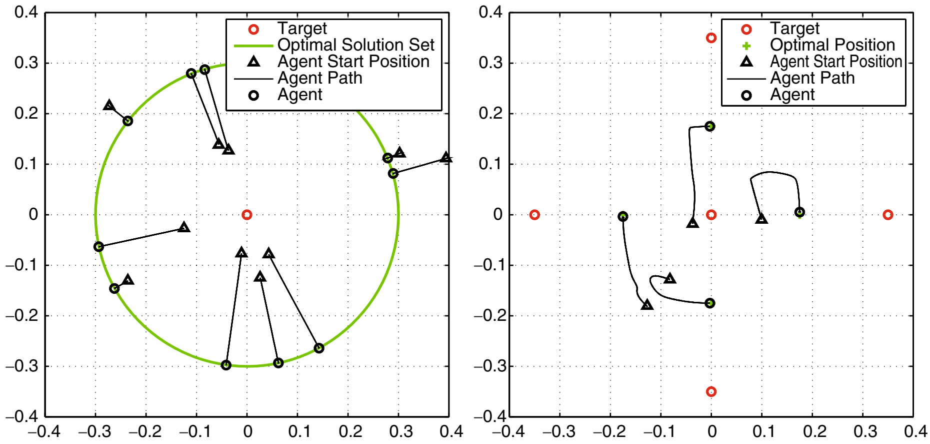

Fig. 8 (*Left*) A team of ten agents charged with observing a static target converge to a globally optimal arrangement in $\mathbb{R}^2$. In this scenario, the desired standoff distance was 0.30 units with $\epsilon_{ij}^l = \epsilon_{ij}^u = 0.15, \forall i, j$. (*Right*) A team of four agents also converge to a globally optimal arrangement while observing five static targets. In this case, the desired standoff distance was 0.175 units with $\epsilon_{ij}^l = \epsilon_{ij}^u = 0.125, \forall i, j$. In both scenarios, $\Psi(X) = -\lambda_2(L_V(X))$

explore the effects of the interactive control functions on convergence is the focus of continued research on this topic.

10 Conclusions and Future Work

In this paper, we considered an optimization framework for dynamic target tracking with performance guarantees for multi-agent systems. To realize this framework, we introduced the notion of a weighted visibility graph to capture the state of active target tracks as a function of the team's state positional vector, X. Noting that dynamic target tracking lends itself well to a discrete-time framework, we employed standard linearization techniques to define an iterative SDP approach for solving the target tracking problem subject to network connectivity constraints. We also considered a framework extension that ensures a minimal cardinality coverage for each of the observation targets. In cases where communication constraints can be relaxed, we presented a novel SOCP relaxation to the target tracking problem that ensures connectivity of the state-dependent visibility graph while providing a tremendous reduction in computational cost when compared to a standard SDP formulation. In solving said relaxation, the resulting configuration guarantees at least a single team member is tracking each target at all times. Additionally, we included a brief discussion on enforcing inter-agent proximity constraints, which can be readily done using a state-dependent LMI. Lastly, we presented preliminary results on the convergence properties of our discrete-time paradigm with respect to static target arrangements.

It should also be mentioned that the performance of the proposed framework is independent of the agent–target ratio. So long as the team can feasibly track the target set given visibility (e.g. coverage) and network constraints, the resulting problem formulation is guaranteed to yield optimal results over some time step. This highlights the tremendous flexibility of the framework for a diverse range of tracking scenarios.

As a final note, there are obvious areas where this work can be improved. For instance, in some applications, a simple Gaussian potential may not fully capture the desired behavior for agent–target interactions. To address this issue, we are currently considering alternate non-symmetric weight functions. Another obvious improvement to this framework is a bit more challenging. In ensuring k-coverage, we introduce m graphs—each having a combinatoric number of edges. Clearly, such a formulation does not lend itself well to large-scale problems. Formulating a more efficient approach for enforcing said constraints is also a topic of significant interest and will be explored.

Acknowledgements The authors gratefully acknowledge the contribution of National Research Organization as well as Yoonsoo Kim for his thoughtful remarks in response to questions on implementing the iterative SDP approach. Additionally, the authors would like to thank the reviewers for their insightful remarks and suggestions.

References

1. Spletzer, J., Taylor, C.: Dynamic sensor planning and control for optimally tracking targets. Int. J. Rob. Res. **22**(1), 7–20 (2003)
2. Kim, Y., Mesbahi, M.: On maximizing the second smallest eigenvalue of a state-dependent graph laplacian. IEEE Trans. Automat. Contr. **51**, 116–120 (2006)
3. Winfield, A.F.T.: Distributed sensing and data collection via broken ad hoc wireless connected network of mobile robots. In: Parker, G.B.L.E., Barhen, J. (eds.) Distributed Autonomous Robotic Systems, vol. 4, pp. 273–282. Springer, New York (2000)
4. Arkin, R., Diaz, J.: Line-of-sight constrained eploration for reactive multiagent robotic teams. In: AMC 7th International Workshop on Advanced Motion Control, Maribor, 3–5 July 2002
5. Sweeney, J., Brunette, T.J., Yang, Y., Grupen, R.A.: Coordinated teams of reactive mobile platforms. In: Proceedings of the International Conference on Robotics and Automation (ICRA), pp. 99–304. Washington, DC (May 2002)
6. Pereira, G.A.S., Das, A.K., Kumar, V., Campos, M.F.M.: Decentralized motion planning for multiple robots subject to sensing and communication constraints. In: Proceedings of the Second Multi-Robot Systems Workshop, pp. 267–278. Washington, DC (March 2003)
7. Wagner, A.R., Arkin, R.C.: Communication-sensitive multi-robot reconnaissance. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 2480–2487. IEEE, Piscataway (2004)
8. Sweeney, J.D., Grupen, R.A., Shenoy, P.: Active qos flow maintenance in controlled mobile networks. In: Proceedings of the Fourth International Symposium on Robotics and Automation (ISRA). IEEE, Queretaro (2004)
9. Powers, M., Balch, T.: Value-based communication preservation for mobile robots. In: 7th International Symposium on Distributed Autonomous Robotic Systems, Toulouse, 23–25 (June 2004)
10. Hsieh, M.A., Cowley, A., Kumar, V., Taylor, C.J.: Towards the deployment of a mobile robot network with end-to-end performance guarantees. In: International Conference on Robotics and Automation (ICRA) 2006, Orlando (April 2006)
11. LaValle, S., Gonzalez-Banos, H., Becker, C., Latombe, J.: Motion strategies for maintaining visibility of a moving target. In: Proceeding of the IEEE Int. Conference on Robotics and Automation. pp. 731–736, Albuquerque (April 1997)

12. Stamos, I., Allen, P.: Interactive sensor planning. In: Computer Vision and Pattern Recognition Conference, pp. 489–495. Santa Barbara (June 1998)
13. Fabiani, P., Gonzalez-Banos, H., Latombe, J., Lin, D.: Tracking a partially predictable object with uncertainties and visibility constraints. J. Auton. Robots **38**(1), 31–48 (2001)
14. Liu, Z., Ang Jr. M.H., Seah, W.K.G.: A potential field based approach for multi-robot tracking of multiple moving targets. In: Proc. 1st International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, Manila (March 2003)
15. Jung, B.: Cooperative target tracking using mobile robots. Ph.D. dissertation, University of Southern California, Los Angeles (2005)
16. Krishna, K.M., Hexmoor, H., Sogani, S.: A t-step ahead constrained optimal target detection algorithm for a multi sensor surveillance system. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robot and Systems, pp. 1840–1845. Edmonton (October–November 2005)
17. Mirzaei, F.M., Mourikis, A.I., Roumeliotis, S.I.: On the performance of multi-robot target tracking. In: Accepted to IEEE International Conference on Robotics and Automation (ICRA) 2007, Rome (April 2007)
18. Shucker, B., Bennett, J.K.: Target tracking with distributed robotic macrosensors. In: Proceedings of MILCOM 2005, Atlantic City (October 2005)
19. Stroupe, A., Balch, T.: Value-based observation with robot teams (vbort) for dynamic targets. In: Proceedings of IROS 2003, Las Vegas (September 2003)
20. Gennaro, M.D., Jadbabaie, A.: Decentralized control of connectivity in multi-agent systems. In: Proc. IEEE Conf. on Decision and Control, San Diego (December 2006)
21. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge Unviersity Press, Cambridge (2004)
22. Isler, V., Khanna, S., Spletzer, J., Taylor, C.: Target tracking with distributed sensors: the focus of attention problem. J. Comput. Vis. Image Underst. **100**, 225–247 (1988) (Special Issue on Attention and Performance in Computer Vision)
23. Mourikis, A.I., Roumeliotis, S.I.: Optimal sensing strategies for mobile robot formations: resource-constrained localization. In: Proceedings of Robotics: Science and Systems, Cambridge (June 2005)
24. Advanced Optimization Laboratory: Addendum to the SeDuMi User Guide Version 1.1 Advanced Optimization Laboratory. McMaster University. http://sedumi.mcmaster.ca/ (2006)
25. Lofberg, J.: Yalmip: a toolbox for modeling and optimization in matlab. In: Proceedings of the CACSD Conference. Taipei, Taiwan. http://control.ee.ethz.ch/~joloef/yalmip.php (2004)
26. The MOSEK Optimization Tools Version 3.2 (Revision 8) User's Manual and Reference. MOSEK ApS. http://www.mosek.com (2008)